

REMARKS

Claim Rejections

Claims 1-3, 5-12 and 14 are rejected under 35 U.S.C. §103(a) as being unpatentable over Sidana (6,571,295) in view of Boys (6,516,340).

Drawings

It is noted that the Examiner has accepted the drawings as originally filed with this application.

New Claims

By this Amendment, Applicant has canceled claims 1-3, 5-12 and 14, and has added new claims 15-30 to this application. It is believed that the new claims specifically set forth each element of Applicant's invention in full compliance with 35 U.S.C. § 112, and define subject matter that is patentably distinguishable over the cited prior art, taken individually or in combination.

- 1. In response to the rejections discussed in the paragraph beginning from "As per claims 1 and 8" on page 3 of the outstanding Office Action.**

On page 3 of the outstanding Office Action, the Examiner states:

As per claims 1 and 8 , Sidana disclosed a system and method for providing online web page annotations, or reading records, allowing a number of users to process annotations, or reading records, including displaying a web page and related annotations, or reading records, in a browser of an end user, the related annotations, or reading records, distinguished by the end user's identification data (See Sidana, Column 11, Claim 11, and Column 12, Claim 14), receiving annotations, or reading records, entered by the end users through the annotation, or reading record, function on the web page (See Sidana, Figure 7, and Column 7, lines 13-41), and storing the users' identification

information and related annotated web pages (See Sidana, Column 12, Claim 14). Sidana does not disclose expressly providing a number of annotation, or reading record, functions on the web page and instantly displaying the annotations, or reading records, on the web page. Boys discloses annotation tools, equivalent to reading record functions (See Boys, Column 6, lines 34-43). Boys also discloses displaying the annotations on the web page (See Boys, Column 8, lines 22-24). Sidana and Boys are analogous art because they are from the same field of endeavor of annotating or modifying the contents of web documents. At the time of the invention it would have been obvious to a person of ordinary skill in the art to include the annotation tools and instant display of modified web pages of Boys with the system and method of processing documents of Sidana. The motivation for doing so would have been to enhance and store selected web pages and to provide the content to users (See Boys, Column 2, lines 42-53). Therefore, it would have been obvious to combine Boys and Sidana for the benefit of enhancing and storing selected web pages and providing the content to users to obtain the invention as specified in claims 1 and 8. (Emphasis added).

The content of the cited prior art referenced by the Examiner is listed below.

(a) Sidana states, column 11, Claim 11:

11. A computer system, comprising:
a remote website;
a web page at the remote website;
a local browser providing access to the web page;
means accompanying the browser in the computer system for web page annotations, where the means provide for adding, editing and deleting the web page annotations, storing the annotations at a location remote from the local browser and annotations display options. (Emphasis added).

(b) Sidana states, column 12, Claim 14:

14. The computer system of claim 12, wherein each of the web page annotations further includes a name portion indicating a user that created the annotation.

(c) Sidana states, column 7, lines 13-41 and Figure 7:

FIG. 6(b) shows an exemplary format of a data structure in memory 104 of redirector 172 for storing display option values. The data structure preferably includes a flag indicting whether the user wishes to display annotations. (For example, annotations are displayed in area 712 of FIG. 7). This flag is initially set to "true/yes". If the flag is false, redirector 172 will not generate HTML corresponding to area 712. The flag is turned off or on by the user clicking on buttons 720 and 730 of FIG. 7. This user action causes a form to be sent to redirector 172 which sends augmented HTML (with or without annotations) in accordance with the new flag value.

FIG. 7 shows a document displayed on the display screen by the browser in accordance with the augmented HTML 180. The augmented document includes a title 702, a URL of the redirector 704, a button area 706, a URL of the document being displayed 708, the body of the original document 710 (with any internal URLs modified to point to the redirector), and an annotation display area 712. Augmented HTML 180 describes the general appearance and placement of title 702, URL of the redirector 704, button area 706, URL of the document being displayed 708, the body of the original document 710, and annotation area 712 in accordance with the augmented HTML. Each of these portions of display 700 is generated by browser 170 in accordance with augmented HTML 180. The remainder of the display 700 is generated by whatever browser 170 is being used and may vary depending on the manufacturer of browser 170. (Emphasis added).

(d) Boys states, column 6, lines 34-43:

A button labeled Text is provided and adapted to allow a lecturer to insert text wording into any place on a cached WEB page, hereinafter termed a lecture page, being

annotated. This capability includes insertion of paragraphs or text blocks, margin notes, footnotes, annotations that are coded to open a second window containing the actual text, information balloons, and so on. In one embodiment, an HTML text editor may be invoked to actually enable altering of the original text on the page. Annotation tools also allow font selection and other text effects (bold, italic, underline).

(e) Boys states, column 8, lines 20-27:

As lecture pages are streamed to recipients, the lecturer may discuss them in the chat and hold discussions in a follow-the leader format. In this way, the lecturer may conduct a lecture without necessarily editing the WEB pages. Simple text instruction appearing in recipients chat windows would direct recipients to which portions of a lecture page to pay attention to and which portion to disregard.

(f) The statement in Column 2, lines 42-53 by Boys:

In preferred embodiments the teacher-author stations are Internet-capable and the lecture-authoring software cooperates with browser software to enable the teacher-authors to search for and browse candidate WEB pages, to enhance and store selected ones of such pages found, to arrange stored product in an order of desired presentation, and to upload a finished lecture to the lecture server for storage and eventual provision of the lecture to the lecture client stations according to the pre-determined schedule. Enhancement may include one or more of masking content, text annotation, attaching audio or video files, or adding graphic elements to the selected page. (Emphasis added).

Applicant's Response

As per the above underlined portion of statement (a), the system disclosed by Sidana is used to deal with the "annotation" on a web page.

As per the above underlined portion of statement (c), "the body of the original document 710" and "annotation area 712" are displayed on different areas on a web page individually.

Sidana, column 2, lines 25-27, indicates that "The HTML for the document, the HTML for the input forms, and the HTML for the annotations are collectively termed 'augmented HTML'. Therefore, "the HTML for the document" and "the HTML for the annotations" are separate, constitute "augmented HTML" eventually, and then are displayed in a user-end browser.

Sidana, column 4, lines 13-14, indicates that, "The original HTML for the Web document is not modified by the present invention", and see Sidana, column 2, lines 44-45: "The original document is not modified." Hence, the content of "the original document 710" of the above (c) may not be modified, and will be separate from the content of "annotation area 712".

As for the above statement (d), it indicates that Boys discloses the annotation tools. See Boys, Figure 2, the underlined "A button labeled Text" of the above statement (d) is included in Software 35. Besides, see the underlined portion of the above statement (f). See Boys, Column 4, lines 60-63: "Software 35 is, in a preferred embodiment, a browser plug-in integrates a unique capability of caching and annotation and bundling WEB pages to normal browser function." See Boys, Column 11, lines 66-67, Column 12, lines 1-4: "It will be apparent to one with skill in the art that the enhancements provided by virtue of software instances 35 (lecturer) and 33 (lecture recipient) may be provided in the form of a browser plug-in as described in preferred embodiments, or may be provided as standalone programs that integrate with certain browser functions." The annotation tools disclosed by Boys, and the function of "displaying the annotations on the web page" are through software 35, which is a plug-in software in browser ; in other words, without the plug-in software 35, the browser can not provide the above functions.

As for the statement (e), the Applicant believes that the content displayed in the recipient chat window is not the "annotations". Rather, the content is a reminder to the user.

The present Appendix-2 which is the amended specification in the communication on February 22, 2005 is the following "the present invention".

The "reading records" disclosed by the present invention are different from the "annotations" disclosed by Sidana. The "reading records" can indeed modify

the original content of a web page; that is to say, when a web page being displayed, "reading records" are used to modify the HTML code of related portion of the original content of the web page. (See the present invention, page 5, lines 21-35, page 6, lines 1-35, and page 7, lines 1-2). It is different from the "annotations" disclosed by Sidana that the "annotations" of a web page will be separate from the original content of the web page and both of them will be displayed in a user-end browser individually at the same time. In fact, the reading record functions that disclosed by the present invention depend on neither a plug-in software like "software 35" disclosed by Boys nor an approach via a device like "redirector 172" disclosed by Sidana and can be used to modify the original content of a web page by modifying its related HTML code in a user-end browser directly. (see the present invention, page 5, lines 21-35, page 6, lines 1-35, and page 7, lines 1-2). The present invention provides a plurality of reading record functions on a web page with a plurality of scripts, and the scripts are comprised in the source code of the web page (see the present invention, page 4, lines 7-8, and please consult Appendix 1, page 2, lines 5-7, Appendix 5, 6 in the communication on February 22, 2005.). When the web page is loaded into a user-end browser, the scripts enable the end user to modify the original content of the web page in the user-end browser directly, to insert a web page object on the web page, (See the present invention, page 4, lines 10-11, page 4, lines 16-18, page 5, line 12, page 5, lines 21-35, page 6, lines 1-11, and FIG. 4) to change the display format of a portion of the web page, (See the present invention, page 4, lines 12-13, page 4, lines 19-21, page 5, line 13, page 6, lines 12-35, page 7, lines 1-2, and FIG. 5) and the inserted web page object or the changed display format of the portion can be displayed immediately (See the present invention, page 4, line 14, FIG. 4 and FIG. 5) through the user-end browser interpretation. (Please load the web page, <http://140.115.8.222/readingrecords-1/>, which is similar to the web site <http://140.115.8.222/readingrecords/> mentioned in the communication on February 22, 2005, the image of the web page is shown in Appendix-3, and the source code of the web page is shown in Appendix-4, which is similar to Appendix 5 in the communication on February 22, 2005. The source code of the

web page, Appendix-4, includes a plurality of scripts which can modify the original content of the web page in a user-end browser without any plug-in software like “software 35” disclosed by Boys or an approach via a device like “redirector 172” disclosed by Sidana.)

In comparison with Sidana and Boys, the present invention is more similar to the expertise disclosed by Boys but it doesn't need any plug-in software like “software 35” in a user-end browser. As for the method disclosed by Sidana, “redirector 172” receives a user's requirement by HTML form to modify HTML codes, transmitting and displaying the modified HTML in a user-end browser; it is very much different from the present invention, for the present invention enables a user to operate the reading record functions in the user-end browser directly and displays the related reading records in real time without any approach via a device like “redirector 172”.

To sum up, the present invention is obviously different from the invention of Sidana and Boys. And the difference will be used to modify the Claims of the present invention. (Please see Appendix-1, currently amended claims in the communication this time)

2. In response to the rejections discussed in the paragraph beginning “As per claims 2 and 9” on page 4 of the outstanding Office Action.

On page 3 of the outstanding Office Action, the Examiner states:

As per claims 2 and 9, Sidana and Boys disclose the limitations of claims 1 and 8 as described above. Sidana also discloses that the annotation, or reading record, functions allows the end user to insert a web page object on the web page and immediately display an inserted object on the page by making the related HTML codes of the web page object insert the inserted object in a selected location in real time (See Sidana, Column 6, lines 56-67, and Column 7, lines 1-7).

The content referenced by the Examiner is listed below.

Sidana states, column 6, line 56 through column 7, line 7:

As mentioned above, HTML 184 of the requested document may contain URLs that provide links to other documents in the World Wide Web. When redirector 172 receives HTML 184 for the requested document, it first parses the HTML looking for such URLs (e.g., URLs are sometimes identified by their initial string "http://"). Redirector 172 modifies any URLs found in the unaugmented HTML so that, in the augmented HTML, such URLs point back to the redirector, as described below. Redirector 172 also adds additional HTML to the HTML received from server 174. The additional HTML corresponds to previous annotations made by the user (if the "view annotations" option=yes) and to various buttons and user-input devices that will be displayed with the requested document. Thus, augmented HTML 180: (1) contains the URL of the redirector in place of or concatenated with the internal URLs that were in the original document and (2) contains additional HTML so that the browser will display any previously made annotations and will further display certain user-input devices. (Emphasis added).

Applicant's Response

The Examiner states, "... allows the end user to insert a web page object on the web page and immediately display an inserted object on the page ...", according to the above underlined portion, the function Sidana provides is to "adds additional HTML" ("insert a web page object") on the web page in redirector 172, but what the present invention provides is to "adds additional HTML" ("insert a web page object") on the web page in a user-end browser directly. This is different from that of Sidana. Since the present invention provides a plurality of reading record functions on a web page with a plurality of scripts and the scripts are included in the source code of the web page (See the present invention, page 4, lines 7-8, and please consult Appendix 1, page 2, lines 5-7 in the communication on February 22, 2005), the scripts enable a user to insert a web page object (See the present invention, page 5, lines 21-35, and page 6, lines 1-11, and FIG. 4), or to change the display format of a selected portion on the web page (See the present invention, page 6, lines 12-35, and

page 7, lines 1-2, and FIG. 5) directly through the user-end browser interpretation. As for the expertise disclosed by Boys, it has to install a plug-in software 35 in the user-end browser. (See Boys, Column 4, lines 60-63), and therefore it is not the same as the present invention to insert web page objects on a web page directly without any plug-in software like "software 35" in the user-end browser.

3. In response to the rejections discussed in the paragraph beginning "As per claims 3 and 10" on page 4 of the outstanding Office Action.

On page 4 of the outstanding Office Action, the Examiner states:

As per claims 3 and 10, Sidana and Boys disclose the limitations of claims 1 and 8 as described above. Sidana also discloses that the annotation, or reading record, functions provides a means for the end user to change the format attributes of a selected portion on the web page and the web page will immediately display the changed format on the selected portion by replacing the selected portion with the related HTML codes in real time on the web page (See Sidana, Column 12, Claim 21, Column 10, lines 58-67, and Column 11, lines 1-4).

The content of the cited prior art referenced by the Examiner is listed below.

(a) Sidana states, column 12, Claim 21:

21. A method of annotation using a client computer comprising:
receiving code and content associated with a web page;
transmitting annotations associated with the content; and
receiving augmented code that corresponds at least in part to the transmitted annotations. (Emphasis added).

(b) Sidana states, column 10, line 58 through column 11, line 4:

In summary, the present invention allows a user to enter and store custom information in association with a Web document. The present invention does not require

modification of the browser or the server. A preferred embodiment allows a user to add "annotation" information to a document. The annotation information is stored in the memory of a redirector. Whenever the user asks to view a Web document, the redirector adds HTML corresponding to the annotation information to the HTML for the Web document (when the appropriate display option is selected) and sends the "augmented HTML" to the browser to be displayed. The redirector also modifies all URLs occurring within the HTML for the document so that all URLs in the HTML point back to the redirector. (Emphasis added).

Applicant's Response

According to the underlined portion of the above (a), the method Sidana disclosed is to deal with the annotation on a web page. The "reading records" of the present invention differs from the "annotation" of Sidana because "reading records" will modify the original content of web pages while the "annotation" of Sidana is separate from the original content of web pages. In fact, the reading record functions that disclosed by the present invention depend on neither a plug-in software like "software 35" disclosed by Boys nor an approach via a device like "redirector 172" disclosed by Sidana and can be used to modify the original content of a web page in a user-end browser directly. The present invention provides a plurality of reading record functions on a web page with a plurality of scripts, and the scripts are comprised in the source code of the web page (See the present invention, page 4, lines 7-8, and please consult Appendix 1, page 2, lines 5-7 and Appendix 5, 6 in the communication on February 22, 2005). When the web page is loaded into a user-end browser, the scripts enable the end user to modify the original content of the web page in the user-end browser directly, to insert a web page object on the web page, (See the present invention, page 4, lines 10-11, page 4, lines 16-18, page 5, line 12, page 5, lines 21-35, page 6, lines 1-11, and FIG. 4) to change the display format of a portion of the web page, (See the present invention, page 4, lines 12-13, page 4, lines 19-21, page 5, line 13, page 6, lines 12-35, page 7, lines 1-2, and FIG. 5) and the inserted web page object or the changed display format of the portion can be

displayed immediately (See the present invention, page 4, line 14, FIG. 4 and FIG. 5) through the user-end browser interpretation.

As per the underlined portion of the above (b), the expertise disclosed by Sidana is not the end user but "the redirector 172" to change the format attributes of a selected portion on web pages. As for the expertise disclosed by Boys, it requires a plug-in software (software 35) to operate. As described above, the expertise disclosed by the present invention requires neither a plug-in software like "software 35" by Boys nor a device like "the redirector 172" by Sidana and it can directly modify the format attributes of a selected portion on the web page in a user-end browser.

4. In response to the rejections discussed in the paragraph beginning "As per claims 5 and 11" on page 4 of the outstanding Office Action.

On page 4 of the outstanding Office Action, the Examiner states:

As per claims 5 and 11, Sidana and Boys disclose the limitations of claims 2 and 9 as described above. Sidana also discloses that the inserted object can be a text object (See Sidana, Figure 7, element 712) or a hyperlink object linking to a selected address (See Sidana, Column 11, lines 1-10, Column 7, lines 59-67, and Column 8, lines 1-34), which are two of the possible types presented in claims 5 and 11.

The content of the cited prior art referenced by the Examiner is listed below.

(a) Regarding Figure 7, element 712, Sidana states, column 2, lines 22-27:

The redirector modifies the HTML of the requested document to include any previously added annotations, if the user wants such annotations displayed. The redirector also modifies the HTML so that various input forms for allowing user input (such as 'buttons', text boxes, etc.) will be displayed by the browser. The HTML for the document, the HTML for the input forms, and the HTML for the annotations are collectively termed 'augmented HTML.'

Regarding Figure 7, element 712, Sidana states, column 7, lines 24-30:

FIG. 7 shows a document displayed on the display screen by the browser in accordance with the augmented HTML 180. The augmented document includes a title 702, a URL of the redirector 704, a button area 706, a URL of the document being displayed 708, the body of the original document 710 (with any internal URLs modified to point to the redirector), and an annotation display area 712.

(b) Sidana states, column 11, lines 1-10:

The redirector also modifies all URLs occurring within the HTML for the document so that all URLs in the HTML point back to the redirector. Thus, if a user clicks on a display element while viewing the document corresponding to the augmented HTML, the modified URL causes the browser to direct a request for the requested new document through the redirector. Thus, the redirector will also generate augmented HTML for the new document and the new document can also be annotated.

(c) Sidana states, column 7, lines 59-67 and Column 8, lines 1-34 by Sidana:

As discussed above, some documents on, e.g., the World Wide Web, contain URLs within the document. If a user clicks on a display item corresponding to one of these URLs, browser 170 will request and display the document corresponding to the new URL. As will be understood by persons of ordinary skill in the art, these internal URLs can be represented on a displayed document by, for example, a displayed URL, keyword, graphic (known as an "ismap"), or the like.

The following paragraphs explain in more detail how URLs within a document are modified by redirector 172 and the effect of clicking on displayed elements corresponding to these modified URLs. If, for example, HTML184 for the requested document contains a link to the URL:

http:// machine3.company3.com

redirector 172 replaces this link within augmented HTML 180 with a link of:

`http://machine1.company1.com/annot=yes/machine3.
company3.com`

That is, the two URLs are concatenated.

In this example, the HTML for the requested document includes a link to a document stored at the URL "machine3.company3.com". Redirector 172 replaces the original URL by a URL of the redirector

"machine1.company1.com", along with an indication of the user-selected option "annot=yes" and an indication of the original URL "machine3.company3.com".

When the requested document is displayed by the browser (see FIG. 7), the display will include a display element corresponding to the modified URL (not shown, but, for example, a keyword or a graphic). If the user clicks on the display element corresponding to the modified URL (that previously pointed directly to machine3.company3.com), the browser accesses the redirector instead of the document pointed to by the original URL link (see steps 302, etc. of FIG. 3). Because the replaced URL includes information about the new document to be viewed and the annotation option value (yes), steps 304 and 306 will be omitted and HTML 184 for new a document will be fetched, augmented, and displayed as described above. In the described embodiment, the annotation option defaults to "yes", which indicates that previous and new annotations will be displayed.

Applicant's Response

As per the underlined portion of the above (a), the annotation area of "element 712" is an object of Text box, one of the input forms, created by "the redirector 172", and provides end users to input text related with annotation; therefore, the annotation area of "element 712" is not a text object inserted by end users. And "element 712" can merely receive the inputted text by end users; however, the present invention allows end users to insert an object of HTML text directly in any place of a web page in a user-end browser.

As per the underlined portion of the above (b) (c), "hyperlink objects" are modified by "the redirector", leading the modified URL to cause the browser to

direct a request for the requested new document through the redirector; therefore, the "hyperlink objects" stated by Sidana are not inserted by end users.

5. In response to the rejections discussed in the paragraph beginning "As per claims 6 and 12" on page 5 of the outstanding Office Action.

On page 5 of the outstanding Office Action, the Examiner states:

As per claims 6 and 12, Sidana and Boys disclose the limitations of claims 3 and 10 as described above. Boys also discloses that the functions required to change format attributes or its displayed characteristics to the portion of the web page include adding underline on a selected text, or masking a selected visible object (See Boys, Column 6, lines 34-43). Sidana and Boys are analogous art because they are from the same field of endeavor of annotating or modifying the contents of web documents. At the time of the invention it would have been obvious to a person of ordinary skill in the art to include the underlining of selected text of Boys with the system and method of processing documents of Sidana. The motivation for doing so would have been to enhance selected web pages and to provide the updated content to users (See Boys, Column 2, lines 42-53). Therefore, it would have been obvious to combine Boys and Sidana for the benefit of enhancing selected web pages and providing the updated content to users to obtain the invention as specified in claims 6 and 12.

The content of the cited prior art referenced by the Examiner is listed below.

(a) Boys states, column 6, lines 34-43:

A button labeled Text is provided and adapted to allow a lecturer to insert text wording into any place on a cached WEB page, hereinafter termed a lecture page, being annotated. This capability includes insertion of paragraphs or text blocks, margin notes, footnotes, annotations that are coded to open a second window containing the actual text, information balloons, and so on. In one embodiment, an HTML text editor may be invoked to actually enable altering of

the original text on the page. Annotation tools also allow font selection and other text effects (bold, italic, underline). (Emphasis added).

(b) Boys states, column 2, lines 42-53:

In preferred embodiments the teacher-author stations are Internet-capable and the lecture-authoring software cooperates with browser software to enable the teacher-authors to search for and browse candidate WEB pages, to enhance and store selected ones of such pages found, to arrange stored product in an order of desired presentation, and to upload a finished lecture to the lecture server for storage and eventual provision of the lecture to the lecture client stations according to the pre-determined schedule. Enhancement may include one or more of masking content, text annotation, attaching audio or video files, or adding graphic elements to the selected page.

Applicant's Response

Please refer to Boys, Figure 2, the underlined "A button labeled Text" of the above (a) is comprised in Software 35; as per the underlined portion of the above (b) and Boys, Column 4, lines 60~63: "Software 35 is, in a preferred embodiment, a browser plug-in integrates a unique capability of caching and annotation and bundling WEB pages to normal browser function." And as per Boys, Column 11, lines 66-67, Column 12, lines 1-4: "It will be apparent to one with skill in the art that the enhancements provided by virtue of software instances 35 (lecturer) and 33 (lecture recipient) may be provided in the form of a browser plug-in as described in preferred embodiments, or may be provided as standalone programs that integrate with certain browser functions." We know that the annotation tools disclosed by Boys and the function of "displaying the annotations on the web page" can function through software 35, a plug-in software installed in a user-end browser. Without software 35, a user-end browser fails to provide end users the above functions alone.

The reading record functions disclosed by the present invention require no plug-in software like "software 35" in a user-end browser, and the browser can offer the same functions of software 35.

6. In response to the rejections discussed in the paragraph beginning from "As per claims 7 and 14" on page 5 of the outstanding Office Action.

Beginning on page 5 of the outstanding Office Action, the Examiner states:

As per claims 7 and 14, Sidana and Boys disclose the limitations of claims 1 and 8 as described above. Boys also discloses that the web pages contains education materials (See Boys, Column 1, lines 51-67, and Column 2, lines 1-5). Sidana and Boys are analogous art because they are from the same field of endeavor of annotating or modifying the contents of web documents. At the time of the invention it would have been obvious to a person of ordinary skill in the art to include the educational material of Boys with the system and method of processing documents of Sidana. The motivation for doing so would have been to allow educational providers to prepare lectures and the like from existing Web sources and package them for timed execution such that a number of students may learn on-line according to a predetermined schedule, and, in some cases, in real time (See Boys, Column 2, lines 6-12) Therefore, it would have been obvious to combine Boys and Sidana for the benefit of allowing educational providers to prepare lectures and the like from existing Web sources and package them for viewing by students to obtain the invention as specified in claims 7 and 14.

The content of the cited prior art referenced by the Examiner is listed below.

(a) Boys states, column 1, line 51 through column 2, line 5:

Another problem with the kinds of on-line education offered in the prior art is that teachers must research and prepare the material according to software conventions, which requires knowledge of computer languages. A teacher that cannot prepare the lessons according to required software convention must have a technician prepare the

material and make it accessible. There are many complex routines to deal with for both teachers and students.

As described above, the Internet provides access to information on virtually any subject matter. Similarly, there are many educational Web pages that contain information that would relate to fields covered by institutions of higher learning, high schools, or trade schools. Such WEB pages are often public-domain pages and may generally be copied and used for educational purposes without permission from the creator or host of the page. Virtually any subject that is taught in a learning environment may be found on a WEB page on the Internet. In many instances, there is information available from WEB pages that has not been written in any textbook. Such pages may be accessed via one of a variety of known search tools (browsers) that are also freely available to the public.

(b) Boys states column 2, lines 6-12:

What is clearly needed is an educational system wherein educational providers may prepare lectures and the like from existing WEB sources and package them for timed execution such that a plurality of students may learn on-line according to a pre-determined schedule, and in some cases in real time. Such an educational system would eliminate the need for complicated program configurations and pass word log-ins....

Applicant's Response

The expertise disclosed by Sidana and Boys can be applied to education fields as the present invention; however, as described above, the expertise disclosed by the present invention is indeed different from that of Sidana and Boys.

7. In response to "Response to Arguments" beginning on page 6 of the outstanding Office Action.

Beginning on page 6 of the outstanding Office Action, the Examiner states:

5. Applicant's arguments filed 25 August 2005 have been fully considered but they are not persuasive.

Regarding Applicant's argument on Page 3 of the Instant Amendment stating that Sidana combined with Boys fail to teach providing a plurality of reading records functions on the web page, the Office respectfully disagrees. Boys teaches providing a number of annotation tools, equivalent in functionality to the reading record functions of the Instant Application (See Boys, Column 6, lines 34-43). The annotation tools of Boys allows for the display of the annotations instantly on the web page (See Boys, Column 8, lines 22-24). While Applicant states that displaying the reading records on the web page without modifying the HTML via the redirector is not taught by Sidana or Boys, this limitation is not included in the claim language.

In response to applicant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *in re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, the motivation to combine Boys with Sidana would be to enhance and store selected web pages and to provide the content to the users.

The content of the cited prior art referenced by the Examiner is listed below.

(a) Boys states, column 6, lines 34-43:

A button labeled Text is provided and adapted to allow a lecturer to insert text wording into any place on a cached WEB page, hereinafter termed a lecture page, being annotated. This capability includes insertion of paragraphs or text blocks, margin notes, footnotes, annotations that are

coded to open a second window containing the actual text, information balloons, and so on. In one embodiment, an HTML text editor may be invoked to actually enable altering of the original text on the page. Annotation tools also allow font selection and other text effects (bold, italic, underline).

(b) Boys states, column 8, lines 20-27:

As lecture pages are streamed to recipients, the lecturer may discuss them in the chat and hold discussions in a follow-the leader format. In this way, the lecturer may conduct a lecture without necessarily editing the WEB pages. Simple text instruction appearing in recipients chat windows would direct recipients to which portions of a lecture page to pay attention to and which portion to disregard.

Applicant's Response

Regarding the response to (a), as described above, the annotation tools disclosed by Boys can work through software 35, which is a plug-in software installed in a user-end browser; in other words, without software 35, a user-end browser fails to provide end users the above functions alone. On the contrary, the reading record functions disclosed by the present invention can provide the same functions as those of software 35 only with a browser instead of installing any plug-in software like "software 35" in a browser.

Regarding the statement (b), the applicant thinks that the content displayed in the recipient chat window is not the "annotations". Rather, the content is a reminder to the user.

As for the examiner's statement, "While Applicant states that displaying the reading records on the web page without modifying the HTML via the redirector is not taught by Sidana or Boys, this limitation is not included in the claim language." The applicant will modify Claims to distinguish the expertise disclosed by the present invention from that by Sidana and Boys. (Please see Appendix-1, currently amended claims in the communication this time).

Application No. 09/981,797

Neither Sidana, nor Boys disclose, or suggest a modification of their specifically disclosed structures that would lead one having ordinary skill in the art to arrive at Applicant's claimed structure. Applicant hereby respectfully submits that no combination of the cited prior art renders obvious the new claims.

Summary

In view of the foregoing amendments and remarks, Applicant submits that this application is now in condition for allowance and such action is respectfully requested. Should any points remain in issue, which the Examiner feels could best be resolved by either a personal or a telephone interview, it is urged that Applicant's local attorney be contacted at the exchange listed below.

Respectfully submitted,

Date: May 8, 2006

By:


Bruce H. Troxell
Reg. No. 26,592

TROXELL LAW OFFICE PLLC
5205 Leesburg Pike, Suite 1404
Falls Church, Virginia 22041
Telephone: 703 575-2711
Telefax: 703 575-2707

Appendix-3 2006/4/26

URL: <http://140.115.8.222/readingrecords-1/>

110A			
101A	102A	103A	104A
Insert a HTML text object	Add a deletion line	Store the HTML code related with the Reading Records back to the server	Display the original content

120A	
Reading Records	
The HTML code related with the Reading Records	

130A	
The Gettysburg Address -- Abraham Lincoln	
Four Scores and seven years ago our fathers brought forth this continent a new nation, conceived in liberty, and dedicated to the proposition that all men are created equal.	
Now we are engaged in a great civil war, testing whether that nation, or any nation so conceived and so dedicated, can long endure. We are met on a great battle fields of that war.this is a ...	

101A:

To insert a HTML text object in area 130A. Decide the cursor situation with a mouse click first in area 130A and then press button 101A so as to input a red HTML text object. As soon as "Enter" is pressed, the related HTML code will be displayed in area 120A.

102A:

To add a deletion line on a selected HTML text in area 130A. Select a HTML text in area 130A and press button 102A, the deletion line will appear on the selected HTML text instantly while the related HTML code will be displayed in area 120A.

103A:

As soon as button 103A is pressed, the HTML code related with the reading records will be stored back to the server so that the reading records can reappear afterwards.

104A:

As soon as button 104A is pressed, the HTML code related with the reading records in the server will be deleted; that is to say, the original content will be displayed.

```

<HTML>
<HEAD>
<TITLE>::Reading Records::</TITLE>
<style type="text/css">
5  body, input {
        font: 15px Verdana;
        background: #FFFFFF;
    }
    .fieldWidth{
10      width:150px;
    }
    .fieldWidth2{
        width:220px;
    }
15  .eventList {
        font: 11px Verdana;
        height: 200px;
        width: 375;
        background-color:#FFFFFF;
20  }
</style>
</HEAD>
<BODY topmargin="0" leftmargin="0" scroll=no oncontextmenu="window.event.returnValue=false">
<input type=hidden id='sAddAll' value="">
25 
    
    <font id="tempf">&nbsp;</font>
    <input type="text" id="inputBox" value="" style="width: 1px;position:absolute;z-
    index:1;border:0;color:red;background:white;display:none;" onkeyup="inputBoxKeyup()"
30  onmousedown="window.event.cancelBubble=true" onmouseup="window.event.cancelBubble=true"
    onclick="window.event.cancelBubble=true" ondblclick="window.event.cancelBubble=true">
    <center><font size=4>110A</font></center>
    <table align=center border=1 bordercolor=black width=800>
        <tr>
35      <td width=100% align=center>
            <table border=0 width=100% align=center>
                <tr>
                    <td align=center width=25%>
                        <center><font size=4>101A</font></center>
40      <input type=button onmousedown="window.event.cancelBubble=true"

```

```

onmouseup="window.event.cancelBubble=true" id="InsertTextBtn" value="Insert a HTML text object"
class="fieldWidth2" onclick="insertFootNote()">
    </td>
    <td align=center width=25%>
5        <center><font size=4>102A</font></center>
        <input type=button onmousedown="window.event.cancelBubble=true"
onmouseup="window.event.cancelBubble=true" id="AddDelBtn" value="Add a deletion line"
class="fieldWidth" onclick="addDelLine()">
    </td>
10    <td align=center width=25%>
        <center><font size=4>103A</font></center>
        <input type=button onmousedown="window.event.cancelBubble=true"
onmouseup="window.event.cancelBubble=true" id="CloseBtn" value="Store the HTML code related
with the Reading Records
15 back to the server" class="fieldWidth2" onclick="SaveCons()">
    </td>
    <td align=center width=25%>
        <center><font size=4>104A</font></center>
        <input type=button onmousedown="window.event.cancelBubble=true"
20 onmouseup="window.event.cancelBubble=true" id="OriBtn" value="Display the original content"
class="fieldWidth2" onclick="ShowOri()">
    </td>
    </tr>
    </table>
25 </td>
    </tr>
    </table>
    <br><br>
    <center><font size=4>120A</font></center>
30 <center><font size=4>Reading Records</font></center>
    <table align=center border=1 bordercolor=black width=800>
        <tr>
            <td width=100%>
                <div style="OVERFLOW: auto;width:798;height:130" id="RRDiv">
35            </div>
            </td>
        </tr>
    </table>
    <br><br>
40 <center><font size=4>130A</font></center>
    <table align=center border=1 bordercolor=black width=800>

```

[illegible]


```

num=0
for each fltem in fontCollection
    if left(fltem.id,4)="cons" then
        paraId=fltem.id
5        AllOriHtml(num)=paraId & "^^^^" & window.document.all(paraId).innerHTML
        num=num+1
        ReDim Preserve AllOriHtml(num)
    end if
next
10
if window.document.all("sAddAll").value<>"" then
    waitTemp=AddCons()
    waitTemp=AddReadingRecords()
end if
15
Set downRange=window.document.body.createTextRange
downRange.moveToElementText window.document.all("cons2")
downRange.collapse(true)
editCursorDisplay()
20 srcDLObj="cons2"
End Sub

Sub document_onmousedown()
    fontdown
25 End Sub

Sub fontdown()
    selText=""

30 if trim(window.document.all("inputBox").value)<>"" then
    inputBoxChange
elseif window.document.all("inputBox").style.display="" then
    window.document.all("inputBox").style.display="none"
end if

35
if window.event.srcElement.parentElement.id="MyList" then
    window.editCursor.style.display="none"
    srcDLObj=window.event.srcElement.id
end if

```

```

End Sub

Sub fontup()
    if window.event.srcElement.parentElement.id="MyList" then
5        if window.document.selection.createRange.text<>"" then
            window.editCursor.style.display="none"
            selText=window.document.selection.createRange.text
            Set Range1=window.document.selection.createRange
            Set Range2=window.document.selection.createRange
10            Range1.collapse(true)
            Range2.collapse(false)
            srcStartObj=Range1.parentElement.id
            srcEndObj=Range2.parentElement.id
        else
15            selText=""
            waitTemp=DownPos(window.event.srcElement.id)
            waitTemp=editCursorDisplay()
        end if
    end if
20 End Sub

Sub fontclick()

End Sub

25 Sub fontdblclick()
    if window.event.srcElement.parentElement.id="MyList" then
        if window.document.selection.createRange.text<>"" then
            window.editCursor.style.display="none"
30            srcDLObj=window.event.srcElement.id
            selText=window.document.selection.createRange.text
            Set Range1=window.document.selection.createRange
            Set Range2=window.document.selection.createRange
            Range1.collapse(true)
35            Range2.collapse(false)
            srcStartObj=Range1.parentElement.id
            srcEndObj=Range2.parentElement.id
        end if
    elseif window.event.srcElement.id="editCursor" then

```

```

        if window.document.selection.createRange.text<>"" then
            window.editCursor.style.display="none"
            srcDLObj=window.downRange.parentElement.id
            selText=window.document.selection.createRange.text
5           Set Range1=window.document.selection.createRange
            Set Range2=window.document.selection.createRange
            Range1.collapse(true)
            Range2.collapse(false)
            srcStartObj=Range1.parentElement.id
10          srcEndObj=Range2.parentElement.id
        end if
    end if
End Sub

15 Sub editCursor_ondblclick()
    window.document.all(window.downRange.parentElement.id).ondblclick
    window.event.cancelBubble=true
End Sub

20 Sub addDelLine()
    if selText<>"" then
        if srcStartObj<>srcEndObj then
            msgbox "It can't add a deletion line here!",64,"Warning"
            window.document.selection.empty
25          exit sub
        end if
        if instr(1,lower(window.document.selection.createRange.htmltext),"<img ")<>0 then
            msgbox "It can't add a deletion line here!",64,"Warning"
            window.document.selection.empty
30          exit sub
        end if
        selId=srcDLObj
        PosData=GetSelPos(selId)
        PosTemp=split(PosData,"|||")
35
        sPos=PosTemp(0)

        appId="unl" & window.document.all(selId).step & "_" & selId
        window.document.all(selId).sAdd=window.document.all(selId).sAdd &

```

```

window.document.all(selId).step & "@<s id=""&appId&"">" & selText & "</s>@" & sPos & "|" &
len(selText) & "@"&appId&"" & "^^"

    window.document.all(selId).step=cint(window.document.all(selId).step)+1
    PageChange=true
5    AddReadingRecords()
    eachStep=split(window.document.all(selId).sAdd,"^^")
    tempHtml=window.document.all(selId).innerHTML
    tempHtml=replace(tempHtml,chr(13),"")
    tempHtml=replace(tempHtml,chr(10),"")
10    NewHtml=tempHtml
    for j=0 to cint(window.document.all(selId).step)-2
        addChar=0
        stepCons=split(eachStep(j),"@")
        if j=0 then
15            if stepCons(1) <> "all" then
                if instr(1,stepCons(2),"|")=0 then
                    NewHtml=mid(NewHtml,1,stepCons(2)) & stepCons(1) &
mid(NewHtml,stepCons(2)+1)
                else
20                    tempPos=split(stepCons(2),"|")
                    NewHtml=mid(NewHtml,1,tempPos(0)) & stepCons(1) &
mid(NewHtml,cint(tempPos(0))+1+cint(tempPos(1)))
                end if
            end if
25        else
            for k=0 to j-1
                stepTemp=split(eachStep(k),"@")
                if instr(1,stepCons(2),"|")=0 then
                    Pos2=cint(stepCons(2))
30                else
                    tempPos=split(stepCons(2),"|")
                    Pos2=cint(tempPos(0))
                end if
                if instr(1,stepTemp(2),"|")=0 then
35                    Pos1=cint(stepTemp(2))
                    Pos3=0
                else
                    tempPos=split(stepTemp(2),"|")
                    Pos1=cint(tempPos(0))
40                    Pos3=cint(tempPos(1))

```

```

end if
if Pos1<Pos2 then
    if stepTemp(1)<>"all" then
        if Pos3=0 then
5             tempS=instr(1,stepTemp(1),">")
              tempE=instr(tempS+1,stepTemp(1),"</")
              addChar=addChar+len(stepTemp(1))-
len(mid(stepTemp(1),tempS+1,tempE-tempS-1))+1
              else
10             addChar=addChar+len(stepTemp(1))-Pos3
              end if
            end if
          end if
        next
15      if stepCons(1)<>"all" then
          if instr(1,stepCons(2),"|")=0 then
              Pos1=cint(stepCons(2))
          else
              tempPos=split(stepCons(2),"|")
20             Pos1=cint(tempPos(0))
          end if
          if mid(lcase(NewHtml),Pos1+1+addChar,2)("</" then
              if instr(1,stepCons(2),"|")=0 then
25             unEndTemp=instr(1,mid(NewHtml,stepCons(2)+1+addChar,">")
                          NewHtml=mid(NewHtml,1,stepCons(2)+addChar+unEndTemp)
                          & stepCons(1) & mid(NewHtml,stepCons(2)+1+addChar+unEndTemp)
                          else
                              tempPos=split(stepCons(2),"|")
30             unEndTemp=instr(1,mid(NewHtml,tempPos(0)+1+addChar,">")
                          NewHtml=mid(NewHtml,1,tempPos(0)+addChar+unEndTemp)
                          & stepCons(1) & mid(NewHtml,cint(tempPos(0))+cint(tempPos(1))+1+addChar+unEndTemp)
                          end if
35             else
                if instr(1,stepCons(2),"|")=0 then
                    NewHtml=mid(NewHtml,1,stepCons(2)+addChar) &
stepCons(1) & mid(NewHtml,stepCons(2)+1+addChar)
                else
40             tempPos=split(stepCons(2),"|")
                    NewHtml=mid(NewHtml,1,tempPos(0)+addChar) &

```

```

stepCons(1) & mid(NewHtml,cint(tempPos(0))+cint(tempPos(1))+1+addChar)
                end if
            end if
        end if
5         end if
        next
        window.document.all(selId).innerHTML=NewHtml
        window.document.selection.empty
        selText=""
10    end if
End Sub

Dim tempSrcDLObj
Sub insertFootNote()
15    if downRange="" or window.editCursor.style.display="none" then exit sub
        footNote=""
        insertType="fn"
        tempSrcDLObj=srcDLObj
        window.editCursor.style.display="none"
20    window.inputBox.style.posTop=PosTop-3
        window.inputBox.style.posLeft=PosLeft-3
        window.inputBox.value=""
        window.inputBox.style.display=""
        window.inputBox.focus()
25 End Sub

Sub addFootNote()
    if footNote="" then exit sub
    downRange.move "character",-1
30
    srcDLObj=tempSrcDLObj
    RealPos=GetPos(srcDLObj)
    OriHtml=GetOriHtml(srcDLObj)

35    appId="fno" & window.document.all(srcDLObj).step & "_" & srcDLObj
        window.document.all(srcDLObj).sAdd=window.document.all(srcDLObj).sAdd & "" &
window.document.all(srcDLObj).step & "<span id=" & appId & " border=0 alt=" & footNote & "
style='color:red'>" & footNote & "</span>@" & RealPos & "@" & appId & "" & ""
        PageChange=true

```

```

AddReadingRecords()
footNote=replace(footNote,chr(13),"<br>")
NewHtml=NowHtml(srcDLObj,OriHtml)
window.document.all(srcDLObj).innerHTML=NewHtml
5   window.document.all(srcDLObj).step=cint(window.document.all(srcDLObj).step)+1
    window.editCursor.style.display="none"
    downRange=""
End Sub

10  Function inputBoxKeyup()
    if window.event.keyCode=27 then
        window.document.all("insertTemp").outerHTML=""
        window.inputBox.style.display="none"
        window.inputBox.value=""
15  end if
    if window.event.keyCode=13 then
        window.inputBox.value=window.document.all("insertTemp").innerText
        if window.inputBox.value<>"" then inputBoxChange()
    else
20      if window.document.all("insertTemp") is nothing then
            inputValue=window.inputBox.value
            window.inputBox.style.display="none"
            FontTemp="<font color='red' id='insertTemp' > "&inputValue&"</font><input
type='text' id='inputBox2' value=" style='width:1px;border:0;color:red;background:white;'
25  onkeyup='inputBoxKeyup()' onmousedown='window.event.cancelBubble=true'
    onmouseup='window.event.cancelBubble=true' onclick='window.event.cancelBubble=true'
    ondblclick='window.event.cancelBubble=true'>"
            downRange.pasteHTML FontTemp
            window.setTimeout "inputFocus()",100,"VBScript"
30      else
            inputValue=window.inputBox2.value
            window.inputBox2.value=""

            window.document.all("insertTemp").innerText=window.document.all("insertTemp").innerText &
35  inputValue
            window.inputBox.value=window.document.all("insertTemp").innerText
            window.inputBox2.focus
        end if
    end if
40  End Function

```

```

Sub inputFocus()
    window.inputBox2.focus
End Sub

5  Function inputBoxChange()
    if window.inputBox.value<>"" then
        window.document.all("insertTemp").outerHTML=""
        footNote=window.inputBox.value
        if insertType="fn" then
10         addFootNote()
        elseif insertType="efn" then
            editFootNote()
        end if
        if window.document.all("inputBox2") is nothing then
15         else
            window.document.all("inputBox2").outerHTML=""
        end if
        window.inputBox.value=""
    end if
20 End Function

Sub editFN()
    insertType="efn"
    footNote=""
25    appId=srcDLObj
    fnTemp=window.document.all(appId).alt

    Set tempRange=window.document.body.createTextRange
    tempRange.moveToElementText window.document.all(appId)
30    tempRange.collapse(true)

    window.inputBox.style.posTop=tempRange.offsetTop+16
    window.inputBox.style.posLeft=tempRange.offsetLeft

35    window.inputBox.value=fnTemp
    window.inputBox.style.display=""
    window.inputBox.focus
End Sub

```



```

Sub editFootNote()
    appId=srcDLObj
    if footNote="" then
        msgbox "Please input text!",48,"Warning"
5        srcDLObj="_"
        exit sub
    end if
    window.document.all(appId).alt=footNote
    paraId=window.document.all(appId).parentElement.id
10
    eachStep=split(window.document.all(paraId).sAdd,"^^")
    for i=0 to cint(window.document.all(paraId).step)-1
        stepCons=split(eachStep(i),"@")
        if stepCons(3)=appId then
15            for j=0 to cint(window.document.all(paraId).step)-2
                stepConsTemp=split(eachStep(j),"@")
                if j<i then
                    stepTemp=stepTemp & stepConsTemp(0) & "@" & stepConsTemp(1) &
"@ " & stepConsTemp(2) & "@" & stepConsTemp(3) & "^^"
20                elseif j=i then
                    stepTemp=stepTemp & stepConsTemp(0) & "<img
src='images/small_pencil.gif' id='"&appId&"' border=0 alt='"&footNote&"' style='cursor:hand'
onclick='jumpFN()'>@" & stepConsTemp(2) & @" & stepConsTemp(3) & "^^"
                    elseif j>i then
25                    stepTemp=stepTemp & stepConsTemp(0) & "@" & stepConsTemp(1) &
"@ " & stepConsTemp(2) & @" & stepConsTemp(3) & "^^"
                    end if
                next
            exit for
30        end if
    next
    window.document.all(paraId).sAdd=stepTemp
    PageChange=true
    AddReadingRecords()
35    srcDLObj="_"
End Sub

Function DownPos(SentId)
    Set downRange=window.document.body.createTextRange
40    downRange.moveToPoint window.event.x,window.event.y

```

```

Set tempRange = document.body.createTextRange
tempRange.moveToElementText(document.all(SentId))
downRange.setEndPoint "StartToStart",tempRange
tempClk=downRange.text
5   tempClk=replace(tempClk,chr(10),"")
    tempClk=replace(tempClk,chr(13),"")
    ClkStartPoint=len(tempClk)
    downRange.collapse(false)
End Function

10  Function editCursorDisplay()
    downX=window.downRange.offsetLeft
    downY=window.downRange.offsetTop

15    window.editCursor.style.display="none"
    PosTop=window.downRange.offsetTop+window.document.body.scrollTop
    PosLeft=window.downRange.offsetLeft+window.document.body.scrollLeft
    window.editCursor.style.posTop=PosTop
    window.editCursor.style.posLeft=PosLeft-5
20    window.editCursor.style.display=""
    window.editCursor.src="images/edit_cursor.gif"
End Function

Function GetSelPos(SentId)
25    Set selectR1=window.document.selection.createRange
    Set selectR2=window.document.selection.createRange
    selectR1.collapse(True)
    selectR2.collapse(False)

30    Set tempR1=window.document.body.createTextRange
    tempR1.moveToElementText(window.document.all(SentId))
    tempR1.collapse(true)
    l=1
    selectR1.setEndPoint "StartToStart",tempR1
35    tempText=replace(selectR1.text,chr(10),"")
    tempText=replace(tempText,chr(13),"")
    R1Len=len(tempText)
    R1CheckLen=len(selectR1.text)
    selectR1.collapse(false)

```

```

selectR1.move "character",-cint(R1Len)
selectR1.setEndPoint "StartToStart",tempR1
if len(selectR1.htmlText)=0 then l=0
selectR1.collapse(false)
5  do while(l=1)
        selectR1.move "character",-1
        selectR1.setEndPoint "StartToStart",tempR1
        if len(selectR1.htmlText)=0 then l=0
        selectR1.collapse(false)
10  R1Len=R1Len+1
loop

Set tempR2=window.document.body.createTextRange
tempR2.moveToElementText(window.document.all(SentId))
15 tempR2.collapse(true)
l=1
selectR2.setEndPoint "StartToStart",tempR2
tempText=replace(selectR2.text,chr(10),"")
tempText=replace(tempText,chr(13),"")
20 R2Len=len(tempText)
selectR2.collapse(false)
selectR2.move "character",-cint(R2Len)
selectR2.setEndPoint "StartToStart",tempR2
if len(selectR2.htmlText)=0 then l=0
25 selectR2.collapse(false)
do while(l=1)
        selectR2.move "character",-1
        selectR2.setEndPoint "StartToStart",tempR2
        if len(selectR2.htmlText)=0 then l=0
30 selectR2.collapse(false)
        R2Len=R2Len+1
loop

eachStep=split(window.document.all(SentId).sAdd,"^^")
35 for i=0 to int(window.document.all(SentId).step)-2
        stepCons=split(eachStep(i),"@")
        if stepCons(1)<>"all" then
                if window.document.all(stepCons(3)) is nothing then
                        elseif lcase(left(stepCons(3),3))="fno" then

```

```

        Set checkR1=window.document.body.createTextRange
        checkR1.moveToElementText(window.document.all(stepCons(3)))
        checkR1.collapse(true)
        checkR1.setEndPoint "StartToStart",tempR1
5      if len(checkR1.text)<=R1CheckLen then
            R1Len=R1Len-1
            R2Len=R2Len-1
        end if
    end if
10  end if
    next

    for i=0 to ubound(AllOriHtml)-1
        temp=split(AllOriHtml(i),"^^^^")
15      if temp(0)=SentId then
            tempHtml=temp(1)
            exit for
        end if
    next
20  tempHtml=replace(tempHtml,chr(13),"")
    tempHtml=replace(tempHtml,chr(10),"")
    window.document.all(SentId).innerHTML=tempHtml

    selectR1.moveToElementText(window.document.all(SentId))
25  selectR1.collapse(true)
    selectR1.move "character",R1Len
    selectR2.moveToElementText(window.document.all(SentId))
    selectR2.collapse(true)
    selectR2.move "character",R2Len
30

    FDelete=false
    LDelete=false

    if lcase(right(window.document.selection.createRange.htmltext,4))="<br>" then selectR2.move
35  "character",-1
        OriL=len(window.document.all(SentId).innerHTML)
        selectR1.pasteHtml "|#tempS#|"
        if len(window.document.all(SentId).innerHTML)-OriL=14 then FDelete=true
        OriL=len(window.document.all(SentId).innerHTML)

```

```

selectR2.pasteHtml "|#tempE#|"
if len(window.document.all(SentId).innerHTML)-OriL=14 then LDelete=true

    tempHtml=window.document.all(SentId).innerHTML
5    if FDelete then tempHtml=replace(tempHtml,"&nbsp;|#tempS#|","|#tempS#|")
    if LDelete then tempHtml=replace(tempHtml,"&nbsp;|#tempE#|","|#tempE#|")
    tempHtml=replace(tempHtml,chr(13),"")
    tempHtml=replace(tempHtml,chr(10),"")

10    sPos=instr(1,tempHtml,"|#tempS#|")-1

    tempHtml=replace(tempHtml,"|#tempS#|","")
    window.document.all(SentId).innerHTML=tempHtml
    tempHtml=window.document.all(SentId).innerHTML
15    tempHtml=replace(tempHtml,chr(13),"")
    tempHtml=replace(tempHtml,chr(10),"")
    ePos=instr(1,tempHtml,"|#tempE#|")-1

    tempHtml=replace(tempHtml,"|#tempE#|","")
20    window.document.all(SentId).innerHTML=tempHtml

    GetSelPos=sPos & "|||" & ePos
End Function

25 Function GetPos(SentId)
    Set tempR1=window.document.body.createTextRange
    tempR1.moveToElementText(window.document.all(SentId))
    tempR1.collapse(true)
    l=1
30    window.downRange.setEndPoint "StartToStart",tempR1

    window.downRange.select

    R1Len=len(window.downRange.text)-10
35    R1CheckLen=len(window.downRange.text)
    window.downRange.collapse(false)
    window.downRange.move "character",-cint(R1Len)
    window.downRange.setEndPoint "StartToStart",tempR1
    if len(window.downRange.htmlText)=0 then l=0

```

```

window.downRange.collapse(false)
do while(l=1)
    window.downRange.move "character",-1
    window.downRange.setEndPoint "StartToStart",tempR1
5    if len(window.downRange.htmlText)=0 then l=0
    window.downRange.collapse(false)
    R1Len=R1Len+1
loop

10    eachStep=split(window.document.all(SentId).sAdd,"^^")
    for i=0 to int(window.document.all(SentId).step)-2
        stepCons=split(eachStep(i),"@")
        if stepCons(1)<>"all" then
            if window.document.all(stepCons(3)) is nothing then
15                elseif lcase(left(stepCons(3),3))="fno" then
                    Set checkR1=window.document.body.createTextRange
                    checkR1.moveToElementText(window.document.all(stepCons(3)))
                    checkR1.collapse(true)
                    checkR1.setEndPoint "StartToStart",tempR1
20                if len(checkR1.text)<R1CheckLen then
                    R1Len=R1Len-1
                    R2Len=R2Len-1
                end if
            end if
        end if
25    end if
next

    for i=0 to ubound(AllOriHtml)-1
        temp=split(AllOriHtml(i),"^^^^^")
30        if temp(0)=SentId then
            tempHtml=temp(1)
            exit for
        end if
    next
35    window.document.all(SentId).innerHTML=tempHtml

    window.downRange.moveToElementText(window.document.all(SentId))
    window.downRange.collapse(true)
    window.downRange.move "character",R1Len

```

```

        if window.downRange.parentElement.id="MyList" then
            window.downRange.move "character",1
            window.downRange.move "character",-1
5        end if

        window.downRange.pasteHtml "##temp##"
        tempHtml=window.document.all(SentId).innerHTML

10        tempHtml=replace(tempHtml,chr(13),"")
        tempHtml=replace(tempHtml,chr(10),"")

        RealPos=instr(1,tempHtml,"##temp##")-1
        if RealPos<0 then RealPos=0

15        tempHtml=replace(tempHtml,"##temp##","")
        window.document.all(SentId).innerHTML=tempHtml

        GetPos=RealPos
20    End Function

    Function GetOriHtml(SentId)
        OriHtml=window.document.all(SentId).innerHTML

25        OriHtml=replace(OriHtml,chr(13),"")
        OriHtml=replace(OriHtml,chr(10),"")

        GetOriHtml=OriHtml
    End Function

30    Function NowHtml(SentId,OriHtml)
        on error resume next
        eachStep=split(window.document.all(SentId).sAdd,"^^")
        NewHtml=OriHtml

35        NewHtml=replace(NewHtml,chr(13),"")
        NewHtml=replace(NewHtml,chr(10),"")
        for j=0 to cint(window.document.all(SentId).step)-1
            addChar=0
            stepCons=split(eachStep(j),"@")

```

```

    if j=0 then
        if stepCons(1) <> "all" then
            if instr(1, stepCons(2), "|")=0 then
                NewHtml=mid(NewHtml, 1, stepCons(2)) & stepCons(1) &
5      mid(NewHtml, stepCons(2)+1)
            else
                tempPos=split(stepCons(2), "|")
                NewHtml=mid(NewHtml, 1, tempPos(0)) & stepCons(1) &
mid(NewHtml, cint(tempPos(0))+1+cint(tempPos(1)))
10      end if
            end if
        else
            for k=0 to j-1
                stepTemp=split(eachStep(k), "@")
                if instr(1, stepCons(2), "|")=0 then
15      Pos2=cint(stepCons(2))
                else
                    tempPos=split(stepCons(2), "|")
                    Pos2=cint(tempPos(0))
20      end if
                if instr(1, stepTemp(2), "|")=0 then
                    Pos1=cint(stepTemp(2))
                    Pos3=0
                else
25      tempPos=split(stepTemp(2), "|")
                    Pos1=cint(tempPos(0))
                    Pos3=cint(tempPos(1))
                end if
                if Pos1 < Pos2 then
30      if stepTemp(1) <> "all" then
                    if Pos3=0 then
                        tempS=instr(1, stepTemp(1), ">")
                        tempE=instr(tempS+1, stepTemp(1), "</")
                        addChar=addChar+len(stepTemp(1))-
35      len(mid(stepTemp(1), tempS+1, tempE-tempS-1))+1
                    else
                        addChar=addChar+len(stepTemp(1))-Pos3
                    end if
                end if
            end if
40      end if

```



```

next
if stepCons(1) <> "all" then
    if instr(1, stepCons(2), "|") = 0 then
        Pos1 = cint(stepCons(2))
    5         else
            tempPos = split(stepCons(2), "|")
            Pos1 = cint(tempPos(0))
        end if
        if mid(lcase(NewHtml), Pos1 + 1 + addChar, 2) = "</" then
            if instr(1, stepCons(2), "|") = 0 then
                unEndTemp = instr(1, mid(NewHtml, stepCons(2) + 1 + addChar, ">")
                NewHtml = mid(NewHtml, 1, stepCons(2) + addChar + unEndTemp) &
10         stepCons(1) & mid(NewHtml, stepCons(2) + 1 + addChar + unEndTemp)
            else
                tempPos = split(stepCons(2), "|")
                unEndTemp = instr(1, mid(NewHtml, tempPos(0) + 1 + addChar, ">")
                NewHtml = mid(NewHtml, 1, tempPos(0) + addChar + unEndTemp) &
15         stepCons(1) & mid(NewHtml, cint(tempPos(0)) + cint(tempPos(1)) + 1 + addChar + unEndTemp)
            end if
        else
            if instr(1, stepCons(2), "|") = 0 then
                NewHtml = mid(NewHtml, 1, stepCons(2) + addChar) & stepCons(1) &
20         mid(NewHtml, stepCons(2) + 1 + addChar)
            else
                tempPos = split(stepCons(2), "|")
                NewHtml = mid(NewHtml, 1, tempPos(0) + addChar) & stepCons(1) &
25         mid(NewHtml, cint(tempPos(0)) + cint(tempPos(1)) + 1 + addChar)
            end if
        end if
    end if
30     end if
end if
next
NowHtml = NewHtml
End Function
35
Function jumpFN()
    srcDLObj = window.event.srcElement.id
    editFN()
    window.event.cancelBubble = true
40 End Function

```

```

Function fnDL()
    if window.event.button="1" then
        srcDLObj=window.event.srcElement.id
5        window.document.selection.empty
        window.event.cancelBubble=true
    end if
End Function

10 Function AddReadingRecords()
    NowRRCons=OriRRConsF
    for each fltem in fontCollection
        if left(fltem.id,4)="cons" then
            if window.document.all(fltem.id).sAdd<>"" then
15                ConsTemp1=split(window.document.all(fltem.id).sAdd,"^^")
                l=1
                num=0
                do while(l=1)
                    ConsTemp2=split(ConsTemp1(num),"@")
20                    NewCons="<tr>"
                    IDTemp=split(ConsTemp2(3),"_")
                    HCons=replace(ConsTemp2(1),"<","&lt;")
                    HCons=replace(HCons,">","&gt;")
                    NewCons=NewCons & "<td align=left><font size=2>" & HCons &
25    "</font></td></tr>"

                    NowRRCons=NowRRCons & NewCons

                    num=num+1
                    if ConsTemp1(num)="" then l=0
30                loop
            end if
        end if
    next
    window.document.all("RRDiv").innerHTML=NowRRCons & "</table>"
35 End Function

Sub ShowOri()
    downRange=""
    window.document.all("RRDiv").innerHTML=OriRRCons

```

```

for each fltem in fontCollection
    if left(fltem.id,4)="cons" then
        for i=0 to ubound(AllOriHtml)-1
            temp=split(AllOriHtml(i),"^^^^")
5           if temp(0)=fltem.id then
                fltem.innerHTML=temp(1)
                fltem.step="1"
                fltem.sAdd=""
            end if
10          next
        end if
    next
    window.document.selection.empty
    window.open "delRecords.asp"
15 End Sub

Function SaveCons()
    dim ds
    Set ds=CreateObject("RDS.DataSpace")
20    dim ServerRds
    Set ServerRds=ds.CreateObject("ServerRds.ServerRdsClass","http://140.115.8.222")

    ConsApp=""
    for each fltem in fontCollection
25        if left(fltem.id,4)="cons" then
            paraId=fltem.id
            sAddTemp=window.document.all(paraId).sAdd
            if sAddTemp<>"" then
                L1Temp=paraId
30                sAddPer=split(sAddTemp,"^^")
                for j=0 to cint(window.document.all(paraId).step)-2
                    sAddCons=split(sAddPer(j),"@")
                AppId=sAddCons(3)
                if instr(1,sAddCons(2),"|")>0 then
35                    OriP=split(sAddCons(2),"|")
                    Pos1=OriP(0)
                    Pos2=OriP(1)
                else
                    Pos1=sAddCons(2)

```

```

Pos2=sAddCons(2)
end if

InsertText=sAddCons(1)
5
ConsApp=ConsApp & "RRecords||noclass||RRecords||" & cstr(L1Temp) &
"||00||00||00||00||00||00||" & cstr(AppId) & "||" & cint(Pos1) & "||" & cint(Pos2) & "||" & "" & "||" &
cstr(InsertText) & "^^"
next
10 end if
end if
next
if ConsApp<>"" then ServerRds.ConsApp cstr(ConsApp)
End Function
15
Function AddCons()
sAddCons=split(window.document.all("sAddAll").value,"^^")
l=1
num=0
20 do while(l=1)
sAddSent=split(sAddCons(num),"@")
SentId=sAddSent(0)
if window.document.all(SentId) is nothing then
else
25 SentSAdd=window.document.all(SentId).step & "@" & sAddSent(1) & "@" &
sAddSent(2) & "@" & sAddSent(3)
window.document.all(SentId).step=cint(window.document.all(SentId).step)+1
window.document.all(SentId).sAdd=window.document.all(SentId).sAdd & SentSAdd
& "^^"
30 end if
num=num+1
if sAddCons(num)=" " then l=0
loop
35 for each fltem in fontCollection
if left(fltem.id,4)="cons" then
if fltem.sAdd<>"" then
OriHtml=GetOriHtml(fltem.id)
NewHtml=NowHtml(fltem.id,OriHtml)
40 window.document.all(fltem.id).innerHTML=NewHtml

```

```
        end if
    end if
next
End Function
5  </Script>
```